

**IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
SHERMAN DIVISION**

MOTIO, INC.,

Plaintiff,

v.

**BSP SOFTWARE LLC, and
BRIGHTSTAR PARTNERS, INC.,**

Defendants.

**CIVIL ACTION No. 4:12-CV-00647-ALM
Hon. Amos L. Mazzant**

JURY DEMANDED

**PLAINTIFF MOTIO, INC.'S
MEMORANDUM OF LAW IN OPPOSITION TO
DEFENDANTS' MOTION FOR JUDGMENT
ON THE PLEADINGS UNDER 35 U.S.C. § 101**

Table of Contents

I.	INTRODUCTION	1
II.	FACTUAL BACKGROUND.....	2
A.	Business Intelligence Software And Cognos.....	2
B.	Automatic Version Control Under The ‘678 Patent	3
D.	The Claim Terms At Issue	7
III.	LEGAL STANDARD.....	8
A.	The ‘678 Patent Does Not Claim A Broad Monopoly to Version Control.....	12
B.	The Claimed Invention Provides a Unique Structure With a Specific Method	13
C.	Defendants Reliance on Other District Court Cases is Misplaced	14
IV.	CONCLUSION.....	15

TABLE OF AUTHORITIES

Cases

<i>Alice Corp. v. CLS Bank Int'l</i> 134 S.Ct. 2347 (2014)	9
<i>Bilski v. Kappos</i> , 561 U.S. 593, 601 (2010)	9
<i>Content Extraction and Transmission LLC v. Wells Fargo Bank</i> , 776 F.3d 1343 (Fed. Cir. 2014)	10
<i>DDR Holdings, LLC v. Hotels.com</i> , 773 F.3d 1245 (Fed. Cir. 2014)	11
<i>Smartflash LLC v. Apple Inc.</i> , No. 13 C 447, 2015 WL 661174 (E.D.TX February 13, 2015)	9
<i>Ultramercial, Inc. v. Hulu, LLC</i> , 772 F.3d 709 (Fed. Cir. Nov. 14, 2014)	10

Statutes

35 U.S.C. §101	<i>passim</i>
----------------------	---------------

I. INTRODUCTION

Plaintiff, Motio, Inc. is the owner of U.S. Patent 8,285,678 (“the ‘678 Patent”) which discloses and claims a novel and unique method providing “version control” to business intelligence (“BI”) systems. The concept of version control in BI systems, which is not new, allows a user to maintain all historical versions of content. However, the ‘678 Patent and Motio’s commercial embodiment thereof has a unique and specific structure and operation. As explained below, the ‘678 Patent vastly improves upon what was previously done and provides external automated version control capabilities to a business intelligence system that does not natively provide version control capabilities.

The ‘678 Patent does not fall in to one of the three exceptions to eligible subject matter under 35 U.S.C. §101. The ‘678 Patent integrates an external automated agent into a business intelligence system to perform version control in a very specific way. It does not claim a broad monopoly to the abstract idea of version control, as there were prior methods of version control which are still used today. Thus, it does not “represent the basic tools of scientific and technological work” and is not the basic “building blocks of human ingenuity.”

The broad concept of version control has been known for decades, different types of version control are currently being used, the Defendants obtained a patent on their own version control method and Defendants deny infringement of the ‘678 Patent. Importantly, the ‘678 patented method is not simply a computerized practice of a “fundamental economic practice” that existed prior to the Internet and was not made possible simply because of the Internet or developing computer technology.

II. FACTUAL BACKGROUND

A. Business Intelligence Software And Cognos

Business intelligence (“BI”) software gathers and processes raw data into meaningful and useful information for business purposes. For example, a BI consumer such as a business executive may want information about the seasonal fluctuation of inventory of a specific product or performance of a set of business units over time. BI software can process massive amounts of data stored across multiple datasources into a human-readable visualization such as a “report” or “report output” summarizing the requested information. The report output can contain graphs, charts, lists, figures, and tables to present the information in an easily digestible manner.

IBM Cognos is a BI software product. The Cognos software is installed and runs on one or more servers that users may access over the Internet. To generate a report output, a user of the Cognos system executes a “report specification” a highly specialized and complex artifact.

The report specification, which is one type of “business intelligence artifact” as used in the claims of the patent-in-suit, is a user-authored artifact which is built on top of an associated metadata layer and (in combination with that metadata layer) defines the rules for which data sources should be queried, which queries should be issued, how numerical measures should be summarized across dimensions, and how the resultant information should be displayed (e.g. in a pie chart, a bar chart, in a crosstab, etc.) when the report specification is executed in a business intelligence system. The Patent Trial and Appeal Board in denying institution of *Inter Partes* Review of the patent-in-suit stated that: “[t]he construction of business intelligence artifact is clear from the claims, but it must be pointed out that the term requires the production of output when executed in the business intelligence system.” (Ex. 1, p.8)

A report output is the result of a successful execution of a report specification. Thus, the

report specification is the “back-end” only seen by the report author and the report output is the front-end which is served up to the BI consumer. In a simple example, a report author might create a report specification, that, when executed in the business intelligence system, produces a report output containing visualizations of sales performance for designated product lines, in regions of interest, and during a requested date range. While the above is a simple example, in the real world, a report specification is a complicated and highly technical computer document. Contrary to Defendants’ repeated allegation, a report specification is not a simple business record¹.

B. Automatic Version Control Under The ‘678 Patent

A Cognos environment is comprised of Cognos software which is installed and runs on a set of servers. Users may access the Cognos environment via a web browser across the Internet. Each report specification created is stored in the Cognos environment. A report author may edit report specifications that reside in the Cognos environment. When a report author is done modifying the report specification, the report specification is saved to the Cognos environment.

While a single report specification can be accessed in the Cognos environment and edited by multiple report authors, only the latest version is available. Thus, it is very easy for one report author to overwrite a previous report author’s changes or accidentally introduce errors into a report specification. For example, such errors may result in a report specification which: (1) no

¹ Defendants previously stated in its claim construction brief that “[b]ecause report specifications are dynamic documents that rely on references to underlying sources of data that may change, edits to report specifications, or to the underlying sources of data, can introduce errors.” [dkt #45, p.6] However, for purposes of its current motion, it has changed positions and now refers to report specifications as plain old business records.

longer executes, (2) executes much slower than before, (3) produces incorrect results, and/or (4) produces improperly formatted reports. Given the substantial investment required to create, update, and maintain these report specifications and the dire consequences of introducing errors into a report specification, Motio conceived of a specific method to capture, save, track and receive historical changes made to report specifications. This method provides external automated version control capabilities to a business intelligence system that does not natively provide version control capabilities.

Specifically, during 2005, Motio developed a new type of “version control” product for Cognos which **automatically** captured and maintained all of the historical versions of a report specification in a external version control repository in real-time as the Cognos environment only saves the most recent version. That is, each time a report specification is saved to the Cognos environment, the previous version is replaced by the modified version. Motio’s product allows a report author to view, compare and access historical revisions of that report specification that are only stored in the external version control repository.

Motio filed the parent application to the ‘678 Patent on January 3, 2006, and shortly thereafter Motio released “MotioCI,” the commercial embodiment containing Motio’s **automated** version control. In the ‘678 Patent, as well as Motio’s commercial product, Motio implemented an **“automated agent”** which is external to the Cognos system but interfaces with the Cognos system and provides automated version control. Using an automated agent to monitor the native Cognos environment is significant because the claimed invention provides version control to a BI system which does not provide version control natively. Thus, the

automated agent, which is external to and not part of the native BI system, must continually monitor the BI system for changes².

The external automated agent automatically detects either (1) a request for modification to a report specification or (2) a modification to a report specification. Like prior systems, the modified report specification is saved to the Cognos environment, however, and significantly, the automated agent continually monitors activity in the Cognos environment, automatically detects whether a modification was made to a report specification and causes the modified version to also be saved in a separate, external version control repository along with all previous versions. Thus, a history of all versions of a report specification from the initial version and all subsequent versions along with information about each version, such as the author, day, time, etc... is maintained in the external version control repository.

In the first situation and as shown in Motio's technical tutorial submitted on July 23, 2015, "detecting a request for modification," the following steps occur:

- (1) the report author edits a previously created and stored report specification and clicks "save." This generates a SOAP³ request issued from the web browser to the Cognos server. This SOAP request is the request for modification of the initial report specification;
- (2) the request for modification passes through the Automated Agent;

² Defendants' one cursory reference to an automated agent in its motion (p.10) in which it concludes that it is "just a form of computer software" is contradicted by the opinion of the PTAB which found that none of the prior art reference cited by Defendants disclosed an automated agent. (Ex. 1)

³ SOAP, originally an acronym for Simple Object Access protocol, is a protocol specification for exchanging structured information in the implementation of web services in computer networks. See <http://en.wikipedia.org/wiki/SOAP>

- (3) the Automated Agent allows the request for modification to continue through the Cognos environment where the modified report specification is saved;
- (4) contemporaneously, the Automated Agent analyzes the “request for modification” and determines if the modified report specification needs to be saved to the version control repository;
- (5) if the Automated Agent determines the modified report specification needs to be saved to the version control repository, the Automated Agent obtains the modified report specification from the Cognos environment.

In the second situation, “detecting a modification,” the automated agent periodically interfaces with the Cognos environment to detect any new modified versions of a report specification which may have been missed and to retrieve and store those modified report specifications in the external version control repository and the same method occurs. In both situations, the key is that the automatic agent “automatically” and continually monitors the activity of the business intelligence system and automatically saves each new version of a report specification in the external version control repository without user interaction.

Importantly, prior to the ‘678 Patent, different types version control existed. For example, after Motio filed this case, Defendants filed a Petition for *Inter Partes* Review of the ‘678 Patent with the Patent Office arguing that the ‘678 Patent was invalid based upon numerous prior art references showing various version control methods. After reviewing the ‘678 specification and claims, the Patent Office denied Defendants’ Petition. (Ex. 1)

In distinguishing over the prior version control systems, Motio explained that traditional version control methods were “mechanical in nature, requiring deliberate action by a user to commit a new version to a file to the version control system.” (Ex. 2, pp. 7-10) In discussing the prior CVS and SVN version control systems, Motio explained how those version control systems worked and how they required “explicit user interaction ... to record a new version.” Some

required detailed instructions from the user, and others some implemented “locks” and “merge” instructions and time limitations. (*Id.*) Motio emphasized that the ‘678 claimed method performed version control in the business intelligence environment **“automatically”** and **“without human intervention,”** as modifications to report specifications were detected. (*Id.*)

C. The Claim Terms At Issue

Claims 1 and 4 are the independent claims in the ‘678 Patent. Both claim the method using an automated agent. Claim 1 is version control based upon a “detecting a request” for modification prior to the report specification being saved to the Cognos environment. Claim 4 is version control after the report specification is saved in the Cognos environment. However, for an analysis under 35 U.S.C. §101 Claim 1 is representative of the ‘678 Patent.

Claims 1 of the ‘678 Patent states:

1. In a general purpose computer, a method for providing automatic version control to a business intelligence system, comprising:

creating an initial version of a business intelligence artifact in the business intelligence system, wherein the business intelligence artifact is a user-authored object that produces output when the business intelligence artifact is executed in the business intelligence system, and wherein the business intelligence artifact is selected from the group consisting of: a report specification and an analysis cube;

providing an automated agent that interfaces with the business intelligence system to provide automated version control to the business intelligence artifact; the automated agent independently performing the steps of:

automatically storing the initial version of the business intelligence artifact with a version control system;

detecting a request to the business intelligence system to modify the initial version of the business intelligence artifact to create a subsequent version of the business intelligence artifact that includes the requested modification; and

automatically storing the subsequent version of the business intelligence artifact in the version control system. (Ex. 3, Col. 4-5)

As explained above, the “automated agent” is the structure that implements the claimed version control method by continually monitoring the business intelligence system for changes. The “initial version” is the latest saved version of the report specification that the report author is working on (called “the business intelligence artifact”). The “subsequent version” is the next version the author saves after modifying the initial version. The automated agent is external to the BI system, interfaces with it to monitor native requests in the system, detects a request to modify the initial version and automatically stores the subsequent version in the external version control repository – all without any human intervention.

The specification of the ‘678 Patent describes a system which employs an automated agent to continuously monitor changes to the business intelligence software. (Ex. 3, 4:32-33) Figure 2 of the specification illustrates the relationship between the invention containing the automated agent and the business intelligence system that does not natively provide version control. As shown in Figure 2, the automated agent continually monitors the business intelligence system and automatically stores versions of report specifications in the version control repository.

III. LEGAL STANDARD

A. 35 U.S.C. §101 Generally

35 U.S.C. §101 defines patentable subject matter. The law governing 35 U.S.C. §101 and inventions involving computers has been applied in several Federal Circuit cases subsequent *Alice Corp. v. CLS Bank Int’l*, 134 S.Ct. 2347 (2014). Moreover, in *Smartflash LLC v. Apple Inc.*, No. 13 C 447, 2015 WL 661174 (E.D.TX February 13, 2015) (Ex. 4), this District thoroughly explained 35 U.S.C. §101, *Alice* and those Federal Circuit cases in determining whether a patent contains patentable subject matter under 35 U.S.C. §101.

35 U.S.C. §101 is a “permissive approach to patent eligibility to ensure that ingenuity should receive liberal encouragement.” *Smartflash*, 2015 WL 661174*3 quoting *Bilski v. Kappos*, 561 U.S. 593, 601 (2010). There are three “exceptions” to patentable subject matter – laws of nature, physical phenomena, and abstract ideas – that “represent the basic tools of scientific and technological work.” *Smartflash*, 2015 WL 661174*3 quoting *Alice*, 134 S.Ct. 2347, 2354 (2014). Therefore, patents that claim the “building blocks of human ingenuity” are not patentable. However, patents that “integrate the building blocks into something more” are patentable. *Id.* This distinction is critical and the foundation of 35 U.S.C. §101.

Patent eligibility under 35 U.S.C. §101 involves a two-part test: (1) first the court determines whether the claims are directed toward one of the three ineligible concepts, and if yes, (2) the court determines “whether the patent amounts to more than a patent on the ineligible concept itself.” *Smartflash*, 2015 WL 661174*4 quoting *Alice*, 134 S.Ct. 2347, 2354 (2014).

B. Application Of 35 U.S.C. §101

In *Alice*, the patent claimed the abstract idea of mitigating a settlement risk by using a third-party intermediary which updated records in real-time to assure the parties have sufficient funds for their financial obligations. Because this was done before computers using clearing houses, *Alice* held this was a fundamental economic practice long prevalent in our system of commerce.” *Alice*, 134 S.Ct. at 2354. The Court held the patent invalid because it claimed the broad concept of an everyday economic transaction that occurred before the Internet, and the introduction of a computer using the Internet to do what was done in business before did not transform it into a patent-eligible invention. *Id.* at 2358; *Smartflash*, 2015 WL 661174*4.

In *Ultramercial, Inc. v. Hulu, LLC*, 772 F.3d 709 (Fed. Cir. Nov. 14, 2014) (“*Ultramercial II*”), the patent claimed a method of offering free media in exchange for

watching advertisements on a computer; i.e., the user at a computer clicks on an advertisement and after watching the advertisement, the user can see selected media. *Id.* at 714-715. The court held that the abstract idea of watching an advertisement prior to watching content was done before the Internet (i.e. that is what television has always been), and the Internet is what allowed its implementation on a computer. Thus, the patentee could not monopolize the abstract idea. *Id.* at 717; *Smartflash*, 2015 WL 661174*4.

In *Content Extraction and Transmission LLC v. Wells Fargo Bank*, 776 F.3d 1343 (Fed. Cir. 2014) the patent claimed a method of extracting and digitizing data from hard copy documents, recognizing specific information in the data and storing that information in a data field memory; i.e. automated teller machines that accept checks for deposits. The court held this to be unpatentable because scanners and computers previously extracted data from documents, and doing this from “hard copy documents” was a natural step in computer technology. *Id.* at 1347, 1348; *Smartflash*, 2015 WL 661174*5.

Significantly, these cases are very different than the recent case of *DDR Holdings, LLC v. Hotels.com*, 773 F.3d 1245 (Fed. Cir. 2014). In *DDR Holdings*, the patent claimed a method of generating a web-page combining a host website and a third party merchant. For example, prior to the patented invention, whenever a user clicked on an advertisement, the user would be taken away from the host website and brought to the third-party website. The patent claimed creating a hybrid web-page having the look and feel of host website and information from the third-party advertiser’s website. The court held this to be patent eligible because there was no “brick and mortar” analogy and did not claim “a fundamental economic or longstanding commercial practice which simply added a computer to the Internet to claim a pre-Internet practice. *Id.* at 1256-1259, *Smartflash*, 2015 WL 661174*5. Moreover, the patent described “a specific way to

automate the creation of a composite web page....” and did not preempt every application of the idea of making two web pages look the same. *Id.*

Applying *Alice* and the subsequent Federal Circuit cases, the *Smartfish* court held, like *DDR Holdings*, the patent claimed eligible subject matter. Specifically, the patent in *Smartfish* described data storage and access systems for paying for and downloading digital content such as audio, video, text software games, and other types of games. Although the *Smartfish* court held that this is an abstract idea under the first step of the test, it was patent eligible under the second part of the test -- it amounted to more than just the concept itself and integrated the idea into something more than a broad concept done before computers and the Internet. Specifically, the court held that the claims “recite specific ways of using distinct memories, data types, and use rules... Although in some claims the language is functional and somewhat generic, the claims contain significant limitations on the scope of the invention.” *Id.* at 8.

Importantly, the *Smartfish* court held that the patents “do not apply a known business practice from the pre-Internet world to computers or the Internet.” Rather, it “is a technology that was developed after widespread use of the Internet.” *Id.* The court also deemed it important that it solved a problem that was “unknown in the pre-Internet era” and that “claims do not risk preempting all future inventions related to the broad concept of exchanging access to data for payment on the Internet.” *Id.* at *9. Rather, the court held that the claims “recite specific ways ... beyond the routine use of the Internet....The claims address specific ways [and] include meaningful limitations that ensure the claims are more than a drafting effort to monopolize the abstract idea.” *Id.*

IV. THE ‘678 PATENT CLAIMS A SPECIFIC TYPE OF VERSION CONTROL AND IS THUS PATENT ELIGIBLE

The ‘678 patent claims a specific method of providing implicit, non-optional version

control to a business intelligence system. The invention uses a specific structure in the form of an automated agent that is external to the BI system and without any human intervention (1) interfaces with the BI system, (2) automatically stores the initial version of the report specification in a version control repository, (3) automatically detects a request to modify the initial version of the report specification, and (4) automatically stores the subsequent versions of the report specification in the version control repository.

A. The ‘678 Patent Does Not Claim A Broad Monopoly to Version Control

Version control with a computer has been known and used after the introduction of the Internet. However, the ‘678 claimed version control was new. It is specific as to structure and method of operation. The ‘678 Patent does not simply use a computer to automate what was done pre-Internet or after the introduction of the Internet. Rather, the opposite is true. Like *DDR*, the invention vastly improves upon what was previously done with computers and the Internet. The claimed invention solves a computer specific problem: without any human intervention, provides external automated version control capabilities to a business intelligence system that does not natively provide version control capabilities by monitoring activities in the business intelligence system and detecting requests to modify report specifications. The specific claimed method with the structure disclosed does not “represent the basic tools of scientific and technological work” and is not the basic “building blocks of human ingenuity” *Smartflash*, 2015 WL 661174*3 quoting *Alice*, 134 S.Ct. at 2354.

Rather, the ‘678 claims “integrate the building blocks into something more” which is patentable. *Id.* It integrates an automated agent into a business intelligence system to perform automated version control in the specific way described above. It does not claim a broad monopoly to the abstract idea of version control. There were prior methods of version control

which are still used today many of which were disclosed to the USPTO during prosecution, and many of which were cited by the Defendant as prior art in their petition for *Inter Partes* Review.

Indeed, Avnet obtained its own patent on another method of providing version control to a business intelligence system (Ex. 5) and sued Motio in Illinois in a pending action. Even though Motio's product described above was cited as prior art to Defendants' application, the USPTO nevertheless issued Defendants a patent on its version control. Had the USPTO determined Motio's product broadly disclosed all methods of providing version control in a business intelligence system as argued by Avnet in its motion, it would not have granted Defendants a patent on its own method. Such outcome at the USPTO further demonstrates that the '678 Patent does not claim a broad monopoly to the abstract idea of version control. It adds something more – an automated agent that without any human interaction continuously monitors the business intelligence software to detect any modification and performs version control in a specified manner.

Defendants have also vehemently denied infringement of the '678 Patent alleging their version control does not have an automated agent and does not perform the same method as the '678 claims. (Ex. 6, pp.9-11) Defendants have hired a technical expert to support this allegation. Defendants' non-infringement position further demonstrates that it believes the specific version control method claimed in the '678 Patent does not provide Motio a broad monopoly to version control.

B. The Claimed Invention Provides a Unique Structure With a Specific Method

The '678 patented method including its unique structure is not a fundamental economic practice prior to the Internet or computers and is not a general building block of human ingenuity. The invention was not made possible simply because of the Internet or developing

computer technology. The broad concept of version control has been known for decades, different types of version control are currently being used, the Defendants obtained a patent on their own version control method and the Defendants have denied that its version control product has the structure or operation of the ‘678 claims. The most practical real-world evidence is that the Defendants have received their own patent on their version control product and deny their product has the structure or operation of the ‘678 claimed version control.

Finally, Defendants motion is yet another attempt to invalidate the ‘678 Patent. They initially filed an *Inter Parties* Review at the USPTO submitting over 60 pages of argument on seven separate grounds of invalidity and a 95 paragraph declaration of an expert. After that was denied, Defendants filed the currently-pending Motion for Summary Judgment under 35 U.S.C. §112 for lack of written description. After that briefing was complete, Defendants filed the instant motion. If Defendants argument had any merit, it would have been filed long before. **Consistent with such pattern and practice, Defendants supplemental response to Motio’s Interrogatory served on Motio less than three months before the filing of the instant motion and directed to each basis for Defendants’ allegation that the ‘678 Patent is invalid completely fails to identify 35 U.S.C. §101.** (Ex. 6, pp. 12-13)

C. Defendants Reliance on Other District Court Cases is Misplaced

Defendants reliance on cases that use the word “automatically” is misplaced. *Intellectual Ventures I v. Manufacturers and Traders Trust Co.* involved patents (1) directed to the broad “conventional concept of budgeting” but added a general computer to the claims, (2) directed to the broad concept of the “creation of digital photo albums” and (3) directed to the general concept of “maintaining privacy of customer billing data during electronic purchases.” All of

these general concepts existed before the Internet and the broad claims would “give the patentee a monopoly” on the broad concepts with a computer.

Similarly, in *Hewlett Packard v. ServiceNow, Inc.*, the claims covered the broad concept of monitoring service tickets at a help desk to remind the employees of impending times and actions. The court held that monitoring deadlines and alerting users existed before the computer and adding a general computer to it was not patentable as it would give a monopoly on this broad concept which “describes every conceivable implementation of the abstract idea.”

Synopsys, Inc. v. Mentor Graphics Corp., involved the broad concept of two software instructions within an integrated circuit. The court held that this would cover every integrated circuit and that the same instructions were used pre-integrated circuit and such “examples in the patents were created by the inventors without the use of a computer.”

Finally, in *Enpat, Inc. v. Tenrox, Inc.*, the claims were broad concept claims covering any software-based system for managing projects with a computer. The court held that adding a computer to the claims that had otherwise been done before was not patentable. These cases, with different claims and different technology are not analogous to the present case. As described above, the ‘678 Patent claims are not broad concept claims and simply do not cover every conceivable way to do version control.

IV. CONCLUSION

For the above-mentioned reasons, Motio respectfully requests the Court deny Avnet’s motion.

Dated: July 27, 2015

Respectfully Submitted,

/s/ Jeffrey M. Drake

Jeffrey M. Drake

Lee F. Grossman

MILLER, CANFIELD, PADDOCK AND STONE, P.L.C.

225 West Washington Street, Suite 2600

Chicago, Illinois 60606

Telephone: (312) 460-4200

Facsimile: (312) 460-4288

drakej@millercanfield.com

grossman@millercanfield.com

Kelly J. Kubasta

FERGUSON, BRASWELL & FRASER, PC

2500 Dallas Parkway, Suite 501

Plano, Texas 75093

Telephone: (972) 378-9111

Facsimile: (972) 378-9115

kkubasta@dallasbusinesslaw.com

Counsel for Plaintiff

CERTIFICATE OF SERVICE

I hereby certify that on July 27, 2015, all counsel of record who are deemed to have consented to electronic service are being served with a copy of this document via the Court's CM/ECF system per Local Rule CV-5(a)(3).

/s/ Jeffrey M. Drake

Jeffrey M. Drake